Pandas Assignment

Tasks 1- 4 :

Q1: Import pandas library

[561]:
```
# write your code here ^_^
import pandas as pd
```

Q2: Read instagram_users.csv file

[564]:
```
# write your code here ^_^
df = pd.read_csv('instagram_users.csv')
```

Q3: Print the number of rows and columns contained in the dataset

[567]:
```
# write your code here ^_^
print("Number of Rows & Columns in dataset : ",df.shape)
```

```
Number of Rows & Columns in dataset :  (65326, 18)
```

Q4: Print the size of dataset

[570]:
```
# write your code here ^_^
print("The size of dataset : ",df.size)
```

```
The size of dataset :  1175868
```

Task 5 :

Q5: Print the data type of each column

[573]:
```
# write your code here ^_^
print("Data type for each column : ""\n",df.dtypes)
```

```
Data type for each column :
 pos         int64
flw          int64
flg          int64
bl           int64
pic          int64
lin          int64
cl           int64
cz         float64
ni         float64
erl        float64
erc        float64
lt         float64
hc         float64
pr         float64
fo         float64
cs         float64
pi         float64
class       object
dtype: object
```

Task 6 :

Q6: Print the entire dataset

Note: if your dataset contains more than 60 rows, only the first 5 rows and the last 5 rows will be printed.

[576]:
```
# write your code here ^_^
df
```

[576]:

	pos	flw	flg	bl	pic	lin	cl	cz	ni	erl	erc	lt	hc	pr	fo	cs	pi	class
0	44	48	325	33	1	0	12	0.000000	0.000	0.000000	0.00	0.000	0.000	0.0	0.000	0.111111	0.094985	f
1	10	66	321	150	1	0	213	0.000000	1.000	14.390000	1.97	0.000	1.500	0.0	0.000	0.206826	230.412857	f
2	33	970	308	101	1	1	436	0.000000	1.000	10.100000	0.30	0.000	2.500	0.0	0.056	0.572174	43.569939	f
3	70	86	360	14	1	0	0	1.000000	0.000	0.780000	0.06	0.000	0.000	0.0	0.000	1.000000	5.859799	f
4	3	21	285	73	1	0	93	0.000000	0.000	14.290000	0.00	0.667	0.000	0.0	0.000	0.300494	0.126019	f
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
65321	13	145	642	0	1	0	7	0.461538	0.000	14.270000	0.58	0.000	0.077	0.0	0.000	0.192308	1745.291260	r
65322	652	3000	1300	146	1	1	384	0.000000	0.389	8.520000	0.13	0.000	1.611	0.0	0.000	0.169917	54.629120	r
65323	1500	3700	3200	147	1	1	129	0.000000	0.111	9.390000	0.31	0.722	0.000	0.0	0.056	0.058908	129.802048	r
65324	329	1500	1800	218	1	1	290	0.055556	0.000	6.350000	0.26	0.222	0.500	0.0	0.000	0.103174	53.402840	r
65325	206	659	608	27	1	0	77	0.000000	0.333	25.549999	0.53	0.222	0.222	0.0	0.167	0.017505	604.981445	r

65326 rows × 18 columns

Task 7 & 8 :

Q7: Print the first 5 rows

[579]:
```
# write your code here ^_^
df.head()
```

[579]:

	pos	flw	flg	bl	pic	lin	cl	cz	ni	erl	erc	lt	hc	pr	fo	cs	pi	class
0	44	48	325	33	1	0	12	0.0	0.0	0.00	0.00	0.000	0.0	0.0	0.000	0.111111	0.094985	f
1	10	66	321	150	1	0	213	0.0	1.0	14.39	1.97	0.000	1.5	0.0	0.000	0.206826	230.412857	f
2	33	970	308	101	1	1	436	0.0	1.0	10.10	0.30	0.000	2.5	0.0	0.056	0.572174	43.569939	f
3	70	86	360	14	1	0	0	1.0	0.0	0.78	0.06	0.000	0.0	0.0	0.000	1.000000	5.859799	f
4	3	21	285	73	1	0	93	0.0	0.0	14.29	0.00	0.667	0.0	0.0	0.000	0.300494	0.126019	f

Q8: Print the last 5 rows

[582]:
```
# write your code here ^_^
df.tail()
```

[582]:

	pos	flw	flg	bl	pic	lin	cl	cz	ni	erl	erc	lt	hc	pr	fo	cs	pi	class
65321	13	145	642	0	1	0	7	0.461538	0.000	14.270000	0.58	0.000	0.077	0.0	0.000	0.192308	1745.291260	r
65322	652	3000	1300	146	1	1	384	0.000000	0.389	8.520000	0.13	0.000	1.611	0.0	0.000	0.169917	54.629120	r
65323	1500	3700	3200	147	1	1	129	0.000000	0.111	9.390000	0.31	0.722	0.000	0.0	0.056	0.058908	129.802048	r
65324	329	1500	1800	218	1	1	290	0.055556	0.000	6.350000	0.26	0.222	0.500	0.0	0.000	0.103174	53.402840	r
65325	206	659	608	27	1	0	77	0.000000	0.333	25.549999	0.53	0.222	0.222	0.0	0.167	0.017505	604.981445	r

Task 9 & 10 :

[656]:
```
# write your code here ^_^
print("Total number of null values : ",df.isna().sum().sum())
```

```
Total number of null values :  0
```

Q10: Print the rows that has duplicate values

[652]:
```
# write your code here ^_^
df[df.duplicated()]
```

[652]:

	pos	flw	flg	bl	pic	lin	cl	cz	ni	erl	erc	lt	hc	pr	fo	cs	pi	class
4118	0	0	0	0	0	0	-1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	f
4165	0	30	149	0	1	0	0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	f
5812	0	0	0	0	0	0	-1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	f
6500	0	0	0	0	0	0	-1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	f
8134	0	0	0	0	0	0	-1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	f
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
32745	0	0	0	0	0	0	-1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	f
32747	0	34	7500	0	1	0	0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	f
32748	0	77	7400	0	1	0	0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	f
32749	0	70	7300	0	0	0	0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	f
32764	0	0	0	0	0	0	-1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	f

1082 rows × 18 columns

Task 11 & 12 :

Q11: Remove all duplicate values

[620]:
```
# write your code here ^_^
df = df.drop_duplicates()
```

Q12: Print the number of rows and columns contained in the dataset after removing the duplicate values

[594]:
```
# write your code here ^_^
df.shape
```

[594]:
```
(64244, 18)
```

Task 13 & 14 :

[597]:
```
# write your code here ^_^
df.columns
df.columns = df.columns = ['Num_post','Num_following','Num_followers','Biography_length','Biography_length','Picture_availability',
                           'Link_availability','Average_caption_length','Caption_zero',' Non_image_percentage','Engagement_rate_like',
                           'Engagement_rate_comment','Location_tag_percentage','Average hashtag count','Promotional keywords','Cosine similarity'
                           ,'Post interval','real_fake']
df.columns
```

[597]:
```
Index(['Num_post', 'Num_following', 'Num_followers', 'Biography_length',
       'Biography_length', 'Picture_availability', 'Link_availability',
       'Average_caption_length', 'Caption_zero', '\tNon_image_percentage',
       'Engagement_rate_like', 'Engagement_rate_comment',
       'Location_tag_percentage', 'Average hashtag count',
       'Promotional keywords', 'Cosine similarity', 'Post interval',
       'real_fake'],
      dtype='object')
```

Q14: Change the class's values to real and fake

[600]:
```
# write your code here ^_^

df['real_fake'] = df['real_fake'].apply(lambda x: "real" if x == "r" else "fake")
```


Task 15 & 16 :

[674]:
```
# write your code here ^_^
#df.head(15)
suOffake = (df['real_fake']=='fake').sum()
print (" Number of fake accounts : ",suOffake)
suOfreal = (df['real_fake']=='real').sum()
print (" Number of real accounts : ",suOfreal)
```

```
 Number of fake accounts :  31784
 Number of real accounts :  32460
```

Q16: Print the count, mean, std, min, 25%, 50%, 75% and the max for each column

[677]:
```
# write your code here ^_^
df.describe()
```

[677]:

	Num_post	Num_following	Num_followers	Biography_length	Biography_length	Picture_availability	Link_availability	Average_caption_length	Caption_zero	\t
count	64244.000000	6.424400e+04	64244.000000	64244.000000	64244.000000	64244.000000	64244.000000	64244.000000	64244.000000	
mean	179.545047	1.202470e+03	2297.041732	58.464464	0.959140	0.286673	138.822131	0.254160	0.196484	
std	729.171634	2.188954e+04	2572.939318	64.228211	0.197967	0.452211	216.786922	0.339104	0.253804	
min	0.000000	0.000000e+00	0.000000	0.000000	0.000000	0.000000	-1.000000	0.000000	0.000000	
25%	6.000000	1.310000e+02	403.000000	0.000000	1.000000	0.000000	9.000000	0.000000	0.000000	
50%	31.000000	3.480000e+02	997.000000	34.000000	1.000000	0.000000	48.000000	0.055556	0.083000	
75%	127.000000	8.300000e+02	3500.000000	111.000000	1.000000	1.000000	174.000000	0.444444	0.333000	
max	76200.000000	3.900000e+06	8800.000000	555.000000	1.000000	1.000000	3644.000000	1.000000	1.000000	